

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Simon Mur

**Vgrajen sistem za nadzor oddaljenega
čebelnjaka**

DIPLOMSKO DELO
VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: viš. pred. dr. Robert Rozman

Ljubljana 2016

Rezultati diplomskega dela so intelektualna lastnina avtorja. Za objavljanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika dela:

Danes se avtomatizacija in informatizacija procesov uporabljata praktično na vseh področjih. Čebelarstvo predstavlja vedno bolj pomembno komponento trajnostnega razvoja. V nalogi raziščite možnosti uvedbe daljinskega nadzora nad čebelnjakom. Na tej osnovi realizirajte celostni prototipni sistem, ki bi omogočal samodejno spremljanje stanja v oddaljenih čebelnjakih brez prisotnosti čebelarja. Izberite najbolj pomembne podatke in zgradite sistem, ki bo te podatke zajemal in pošiljal na spletni strežnik ali preko SMS sporočil neposredno uporabniku. Omogočeni naj bodo različni režimi delovanja, ki jih bo mogoče prilagajati željam uporabnika. Prav tako poskušajte zasnovati cenovno konkurenčen sistem, ki bo imel tudi tržno perspektivo. Sistem naj ima dodatno možnost spletne predstavitve shranjenih podatkov.

MENTOR: viš. pred. dr. Robert Rozman

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Simon Mur, z vpisno številko **63070345**, sem avtor diplomskega dela z naslovom:

Vgrajen sistem za nadzor oddaljenega čebelnjaka

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom viš. pred. dr. Robert Rozmana
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 2.3.2016

Podpis avtorja:

Zahvalil bi se svoji družini za podporo in pomoč pri izdelavi diplomske naloge. Prav tako gre zahvala mentorju viš. pred. dr. Robertu Rozmanu za pomoč in potrpežljivost pri moji diplomski nalogi. Zahvalil bi se tudi gospodu Mitji Kodriču iz podjetja IMK ind. elektronika d.o.o., ker mi je podaril merilno celico in s tem omogočil lažjo izdelavo projekta.

Kazalo

Povzetek

Abstract

1	Uvod	1
1.1	Sorodni izdelki	2
1.1.1	SMS Čebelar	2
1.1.2	Telemetrija čebelnjaka - čebelarska tehtnica V3	2
1.1.3	Čebelarska tehtnica TCM-13	3
1.1.4	Oddaljeno opazovanje panja Arnia	4
1.2	Funkcionalnosti sistema za nadzor oddaljenega čebelnjaka	4
1.3	Izbira osnovne plošče: Arduino ali Raspberry Pi	5
1.3.1	Razvojna plošča Arduino	6
1.3.2	Razvojna plošča Raspberry Pi	6
1.3.3	Primerjava sistemov Arduino in Raspberry Pi	6
2	Gradniki in izdelava vgrajenega sistema s tehtnico	9
2.1	Gradniki vgrajenega sistema s tehtnico	9
2.1.1	Osnovna plošča Arduino Mega	9
2.1.2	Merilna celica za težo - Vishay 1022 C3	10
2.1.3	Senzor za temperaturo in vlago Hunwei DHT21	11
2.1.4	Modul za realni čas (RTC) - DS3231	12
2.1.5	Modul za pomnilniške MicroSD kartice	13
2.1.6	LCD prikazovalnik - Blue Serial I2C 20X4	13
2.1.7	GSM ščit - GSM nadgradnja osnovne plošče Arduino	14
2.2	Povezovanje gradnikov v sistem	15

KAZALO

2.2.1	Vezava modula GSM	15
2.2.2	Vezava senzorja za vlago in temperaturo DHT21	15
2.2.3	Vezava modula za realni čas - DS3231	16
2.2.4	Vezava merilne celice teže Vishay 1022 C3	16
2.2.5	Vezava modula za shranjevanje na microSD	18
2.2.6	Vezava prikazovalnika LCD Blue Serial I2C 20X4	18
2.2.7	Napajanje	19
3	Programska oprema	21
3.1	Branje podatkov iz merilne celice ter senzorja za vlago in temperaturo	22
3.2	Funkciji pošiljanja SMS sporočil	22
3.2.1	SMS na zahtevo	22
3.2.2	SMS poslan ob določeni uri	23
3.3	Shranjevanje na microSD kartico	23
3.4	Shranjevanje v podatkovno bazo	24
3.5	Varčevanje z energijo	24
4	Uporaba končnega izdelka v praksi	25
4.1	Montaža v čebelnjak	27
4.2	Spletni prikaz shranjenih podatkov	28
4.3	Testiranje sistema v praksi	31
5	Sklepne ugotovitve	33

Seznam uporabljenih kratic

kratica	angleško	slovensko
SMS	Short Message Service	Sistem kratkih sporočil
SIM	Subscriber Identity Module	modul za identifikacijo naročnika
GSM	Global System for Mobile Communications	Globalni sistem za mobilne komunikacije
AŽ Panj	/	Panj slovenskega čebelarja Antona Žnideršiča. Panj je izdelan na osnovi prvega panja Antona Janše.
RTC	Real Time Clock	Realno časovni modul
SQL	Structured Query Language	Strukturiran povpraševalni jezik
MySQL	Open-source Relational Database Management System	Odprtokodni sistem za upravljanje s podatkovnimi bazami
EEPROM	Electrically Erasable Programmable Read-Only Memory	Električno izbrisljiv programirljiv bralni pomnilnik
SRAM	Static Random-Access Memory	Statičen bralno-pisalni pomnilnik
I2C	Inter-Integrated Circuit	Enostavno komunikacijsko vodilo

Povzetek

Oddaljenost čebelnjakov in različne intenzivnosti paše v poletnih mesecih za čebelarja pomenijo dodatno (velikokrat nepotrebno) vožnjo od čebelnjaka do čebelnjaka. Vse to z namenom ugotavljanja stanja v čebelnjaku. Vožnja prinese za sabo dodatne stroške in predvsem izgubo dragocenega časa.

Cilj diplomske naloge je izdelati cenovno ugoden vgrajen sistem, ki bo poleg funkcionalnosti elektronske tehtnice, preko SMS sporočil pošiljal izbrane podatke uporabniku - čebelarju. Izbrani podatki vključujejo podatke o teži, razliko v teži od prejšnjega merjenja, trenutno vlago in temperaturo. Ti podatki omogočajo načrtovanje nadaljnjega dela v čebelnjaku.

V diplomskem delu je opisan postopek izdelave vgrajenega sistema skupaj z elektronsko tehtnico, od izbire do medsebojne vezave aparaturnih gradnikov, s pripadajočo programsko opremo.

Končni izdelek diplomske naloge je funkcionalna pametna tehtnica s sistemom pošiljanja podatkov preko SMS sporočil in možnostjo vpogleda statistike preko spletne strani. Tehtnica je uporabniku prijazna in enostavna za uporabo ter omogoča bistveno bolj učinkovito upravljanje oddaljenih čebelnjakov.

Abstract

The distance of beehives as well as different intensity in the production of honey during the summer months mean additional (and in many cases unnecessary) drives from beehive to beehive to determine their state. The driving is not only time consuming, but also generates unnecessary financial expenses.

The goal of this work is to produce a cost-efficient integrated system that will not only inform the user, in this case a beekeeper, about the proper functionality of the scale, but will also send an SMS with required information to the user. The data will include beehive's weight, the difference in weight since last check and current humidity and temperature levels. This data enables accurate plan-making regarding further work in the hive.

This work describes the process of integrating an electronic scale into the embedded system, from selection of components to the final product, with the corresponding software.

The final product of this presented work is a smart electronic scale using a SMS messaging system with the option of reviewing past data statistics on a website. The scale is user-friendly and it enables significantly more efficient management of remote beehives.

Poglavje 1

Uvod

V današnjem času, ko sta čas in denar vladarja, si ne moremo privoščiti brezglavega zapravljanja denarja za vožnjo do čebelnjaka le zato, da bi dobili podatke o stanju v njem. Ideja elektronske tehtnice je olajšati delo čebelarju. Saj mu ni potrebna vožnja do čebelnjaka, da bi izvedel, kakšno je stanje. Dovolj je le poslan SMS in podatki, ki jih potrebuje so na voljo. Na trgu že obstaja kar nekaj različic elektronskih tehtnic. Razlikujejo se v izdelavi, natančnosti in seveda v ceni. Cene se gibljejo od 400 € brez DDV naprej.

Namen diplomskega dela je izdelati cenovno ugodno elektronsko tehtnico, ki bi poleg teže sporočala tudi temperaturo, vlago zraka in razliko v teži od prejšnjega merjenja. S temi podatki lahko čebelar vidi stanje v panju. Na primer, če se teža konstantno povečuje, pomeni, da čebele prinašajo večje količine medu. Ko doseže določeno težo, čebelar ve, da je potrebno med iztočiti. V kolikor teža pada dlje časa, potem je potrebno čebele nakrmili.

Preden se lotimo načrtovanja in izdelave našega izdelka je potrebno narediti analizo nekaterih obstoječih rešitev. Tako dobimo podatke o tem, katere funkcije imajo obstoječe tehtnice in kaj so njihove prednosti in slabosti.

Po opravljeni analizi konkurenčnih izdelkov je potrebno opraviti še analizo odprtokodne strojne opreme, ki bi bila primerna za izdelavo našega produkta. Med vsemi najbolj izstopata Arduino in Raspberry Pi. V nadaljevanju je bilo potrebno izbrati še vse ostale aparaturne gradnike, ki jih potrebujemo za delovanje izdelka.

1.1 Sorodni izdelki

V zadnjih letih se je z razvojem tehnologije in dostopnostjo mikrokrmilnikov na trgu zelo povečala ponudba tovrstnih izdelkov. Večina izdelkov, ki so trenutno na trgu, je namenjena končnim uporabnikom, brez možnosti oziroma z minimalnimi možnostmi, ki bi uporabniku omogočale nadgradnjo naprave z nekaj računalniškega znanja.

1.1.1 SMS Čebelar

SMS Čebelar je elektronska tehtnica, namenjena tehtanju donosa v čebeljem panju. Čez dan dvakrat zabeleži težo, temperaturo, vlago, napetost baterije in jakost GSM signala. Vse te podatke pošlje v enem SMS sporočilu na prednastavljeno GSM številko oziroma lahko tudi na številko spletnega strežnika. Obenem pa omogoča tudi ročen vpogled na lokaciji, saj ima vgrajen zaslon. Ima tudi funkcijo SMS alarm, ki pošlje SMS v primeru vandalizma (vklopi se v primeru prekinitve sklenjenega tokokroga senzorja, ki je nameščen v čebelnjaku na primernih mestih).

Tehtnica za delovanje potrebuje delujočo SIM kartico poljubnega operaterja. Vse nastavitve se opravijo prek SMS sporočil.

Na napravi je le ena tipka, ki sproži meritev, v primeru daljšega pritiska pa vstopimo v meni z nastavitvami.

Napaja se s pomočjo 3.6V baterije velikosti "D" in ima kapaciteto 19Ah. S tako baterijo lahko tehtnica deluje nemoteno približno eno leto [2].

1.1.2 Telemetrija čebelnjaka - čebelarska tehtnica V3

Čebelarska tehtnica V3 je nadgrajena in izboljšana od njenih predhodnic V1 in V2, ki sta bili ponujeni na trgu v letu 2013.

Tehtnica pošilja podatke o izmerjeni teži, temperaturi, spremembi temperature glede na prejšnji dan, zračno vlago, napetost akumulatorja, jakost GSM signala ter stanje na računu. Prav tako omogoča SMS alarm. Lahko pa si vse podatke ogledamo tudi na vgrajenem LCD zaslonu.

Tehtnico dobite v kompletu z IZImobil kartico in predlaganim polnjenjem v vrednosti 25€, kar omogoča stranki nemoteno uporabo celo leto, brez skrbi, da po 90 dneh preteče veljavnost kartice.

Tehtnica je v stalni pripravljenosti in omogoča uporabniku prek poslanega praznega SMS pridobitev podatkov iz tehtnice. Prav tako pa omogoča avtomatsko pošiljanje vsako polno uro dneva na do 5 različnih mobilnih števil.

Priložen je tudi program, s katerim opravljamo vse nastavitve. Prek tega programa lahko tudi zaklenemo pošiljanje SMS na 5 števil in tako preprečimo pošiljanje na nepooblašene številke.

Za napajanje potrebuje enosmerno napetost med 11 in 14V. Priporočena je uporaba 12V svinčenega akumulatorja. Da naprava deluje neprestano poskrbi polnjenje prek sončnih celic, ki so priložene.

Ena izmed potencialnih prednosti kompleta V3 je tudi pocinkano robustno ogrodje z vgrajenim senzorjem za tehtanje 200-kilogramskih bremen. Senzor je sicer zaradi natančnosti 100 g umerjen za tehtanje do 100 kg in omogoča, da lahko panj ostane na merilni enoti brez predhodne razbremenitve, ko premični čebelnjak prestavimo na novo lokacijo.

Dodatna varnostna zaščita je zaklep naprave na točno določeno SIM kartico. V primeru odtujitve tako postane naprava neuporabna, saj so vse nastavitve vezane na uporabnika (torej SIM kartico, ki je v napravi) [3].

1.1.3 Čebelarska tehtnica TCM-13

Tako kot predhodnika tudi čebelarska tehtnica TCM-13 pošilja podatke o meritvah teže panja, temperature in vlage zraka. Podatke pošilja prek SMS ali prek spletne aplikacije.

Prav tako omogoča vpogled podatkov kadarkoli na spletni strani www.ecebelar.com, na kateri vidimo podatke prikazane grafično.

Lahko nam sporoča podatke o zaporednih dneh negativnega prirastka in s tem obvesti čebelarja o prenehanju medenja in posledično potrebnih ukrepih.

Ogrodje je robustno in vzdržljivo z vgrajeno merilno celico, ki meri težo do 100 kg, podatka o odstopanju oziroma natančnosti pa ni podanega. Temperaturo meri v razponu od -30 do 50 stopinj Celzija in relativno vlažnost med 10% in 100%. Teža tehtnice znaša 8 kg in ima dimenzije 485x330x80 mm.

Nastavitve lahko enostavno opravimo s SMS sporočili [4].

1.1.4 Oddaljeno opazovanje panja Arnia

Arnia omogoča uporabniku pridobivanje pomembnih informacij o temperaturi, vlagi, teži in aktivnosti čebel (npr. kako pogosto čebele letijo iz panja) [5].

Sistem Arnia je kompleksen in vključuje veliko funkcij, ki so nastavljive prek spletnega vmesnika.

Nekaj ključnih funkcij:

- opozorilo v primeru kraje
- opozorilo v primeru prevrnjenega panja
- temperatura v panju
- nadzor razvoja družine
- nadzor nad aktivnostjo čebel v panju
- zimsko opazovanje

Sistem deluje tako, da tehtnica zbira podatke iz panja in okolice, jih obdela in pošlje prek Arnia omrežja v oblak. Oblak predstavlja v računalniški terminologiji spletno hrambo podatkov, do katerih lahko dostopajo vse v splet povezane naprave in uporabniki. Iz oblaka jih potem uporabnik lahko preko pregledne spletne strani pregleda in si naredi analizo.

Spletna stran omogoča tudi vse nastavitve na tehtnici. V kolikor ima uporabnik več tehtnic, so združene vse v enem samem spletnem vmesniku.

1.2 Funkcionalnosti sistema za nadzor oddaljenega čebelnjaka

Po pregledu sorodnih izdelkov smo ugotovili, kaj potrebuje naš sistem, da bo lahko konkurenčen izdelkom, ki so že na trgu.

Da bo naš izdelek konkurenčen, mora imeti najmanj naslednje funkcionalnosti:

- podatke o teži
- podatke o razliki teže od prejšnje meritve

- podatke o vlagi in temperaturi
- daljšo življenjsko dobo baterije
- enostavno uporabo

Naprava potrebuje neprekinjeno delovanje in ker večina čebelnjakov ni priključenih na električno omrežje, je potrebno urediti napajanje za našo napravo. Zmogljiv 12V akumulator in sončne celice na strehi zlahka rešijo problem.

Ljudje smo si različni, nekateri želijo vedeti stanje v čebelnjaku kadarkoli, drugi so zadovoljni s SMS sporočilom ob koncu dneva. Ravno zato ima naša tehcnica dva načina delovanja.

- Način 1 deluje neprekinjeno ves čas in stranki omogoča, da stanje preveri kadarkoli. Prav tako pa bo sistem pošiljal podatke tudi ob določeni uri popolnoma avtomatsko, tako preko SMS kot na spletno stran, kjer je uporabniku na voljo grafičen prikaz pridobljenih podatkov. Vendar je ta način energijsko zelo potraten in potrebuje močnejši akumulator.
- Način 2 bo omogočal le avtomatsko pošiljanje SMS sporočila in podatkov na internet. Podatki bodo prav tako prikazani grafično na spletni strani. Ta način je bateriji prijaznejši, saj naprava večino časa varčuje energijo in tako omogoča delovanje z manjšo baterijo, s tem pa tudi kompaktnější sistem.

Dodatna prednost našega sistema je nadgradljivost, tako bi uporabniku ob naročilu omogočili izbiro funkcij sistema, ki jih potrebuje za svoj čebelnjak. Obe- nem pa je sistem možno nadgraditi, če se pojavi taka potreba.

1.3 Izbira osnovne plošče: Arduino ali Raspberry Pi

Arduino ali Raspberry Pi povzroča kar precej dileme pri začetnikih. Ti dve napravi sta razmeroma poceni in sta približno enako veliki. Vendar pa sta popolnoma drugačni. Raspberry Pi je mini računalnik, ki ga primarno poganja operacijski sistem Linux (lahko tudi kateri koli drug kompatibilen operacijski sistem), na drugi strani pa je Arduino, mikrokrmilniški sistem brez tipičnega operacijskega sistema. Vsak ima svoj namen [6].

1.3.1 Razvojna plošča Arduino

Arduino je mikrokrmilniški sistem, ki ga lahko programiramo, da služi zelenemu namenu. Namenjen je branju in obdelavi podatkov iz različnih senzorjev. Te podatke lahko pošilja na računalnik ali na katero drugo napravo, ki omogoča obdelavo podatkov. Podatke lahko prikaže tudi prek LED diod ali LCD prikazovalnika.

Arduino je uporabljen v več različnih projektih, od glasbenih aplikacij, 3d tiskalnikov pa do robotike in igralniških dodatkov [6].

1.3.2 Razvojna plošča Raspberry Pi

Raspberry Pi je zmogljivejši sistem, ki se od Arduina precej razlikuje. Ima operacijski sistem Linux, ki je naložen na SD kartici. Na napravi imamo že izhod za zvok, HDMI izhod, RCA video izhod in izhod za internet. Vse to omogoča uporabo sistema Raspberry Pi kot namiznega računalnika (lahko brskamo po internetu, igramo preproste igre, ...). Če priklopimo še tipkovnico in miško imamo zelo poceni računalnik.

Projekti za Raspberry Pi so bolj programsko usmerjeni. Ker sistem poganja preprost Linux operacijski sistem, je večina projektov bolj baziranih na programskih in multimedijskih centrih. Seveda pa lahko nanj priključimo tudi enostavno strojno opremo, ki jo upravljamo prek GPIO pinov [6].

1.3.3 Primerjava sistemov Arduino in Raspberry Pi

Izbira plošče je odvisna od tipa projekta, ki ga imamo v načrtu izdelati, in izkušenj na področju programiranja. Arduino je bolj usmerjen v strojno opremo (kot so LED diode, LCD, različni senzorji z analognim ali digitalnim izhodom, ...), Raspberry Piju pa bolj ustreza programski del, saj je to že majhen, a vseeno zmogljiv računalnik.

Raspberry Pi že vsebuje internet, zvok in sliko (nanj lahko priključimo računalniški monitor). Arduino ima možnost nadgradnje z internetnim ščitom, ki ga dodamo na osnovno ploščo, spraviti zvok na zvočnike ali sliko na monitor pa je že malce težje [6].

Za potrebe našega projekta je najprimernejša izbira osnovna plošča Arduino. Le ta z naborom gradnikov omogoča vse funkcije, ki jih potrebujemo. Osnovna

plošča Arduino deluje zelo dobro z analognimi in digitalnimi izhodi. Namreč vse podatke, ki jih potrebujemo dobimo iz gradnikov v obliki analognih ali digitalnih signalov.

Poglavje 2

Gradniki in izdelava vgrajenega sistema s tehtnico

Za naš projekt smo izbrali osnovno ploščo Arduino Mega 2560 [7], senzor za vlago in temperaturo ter merilno celico za težo. Za pošiljanje podatkov na SMS/internet smo uporabili GSM ščit podjetja Telefonica [13]. Za shranjevanje podatkov lokalno imamo microSD kartico, kopijo podatkov izven naprave pa shranjujemo v bazo MySQL na oddaljenem strežniku.

2.1 Gradniki vgrajenega sistema s tehtnico

2.1.1 Osnovna plošča Arduino Mega

Arduino nam ponuja široko paleto izdelkov, ki so primerni za naš izdelek: modeli Uno, Mega, Mini (z nekaj več znanja elektrotehnike) itd. Izbrati prave osnovne plošče ni tako lahko, tu se moramo vprašati, kaj potrebujemo. Namreč vsi ti moduli se razlikujejo v številu izhodov (oz. vhodov), velikosti spomina (tako za shranjevanje skice, kot za spremenljivke) ...

Na odločitev o izbiri plošče ni vplival mikrokrmilnik ali število vhodov in izhodov, ampak prostor, namenjen shranjevanju našega programa in spomina za spremenljivke. Najprej smo poizkusili z Arduino Uno, vendar pa se je kaj kmalu pojavila težava v pomanjkanju prostora, saj je bilo hitro preseženih 32 KB prostora za program in 2 KB za spremenljivke. Težavo smo rešili z menjavo osnovne

Plošča	Mikrokontroler	Digitalni V/I	Analogni V/I
Uno	ATmega328	14	6
Mega	ATmega2560	54	16
Leonardo	ATmega32u4	20	7

Tabela 2.1: Primerjava Arduino plošč glede na mikrokontroler in število vhodov in izhodov.

Plošča	Flash	SRAM	EEPROM
Uno	32 KB	2 KB	1 KB
Mega	256 KB	8 KB	4 KB
Leonardo	32 KB	2.5 KB	1 KB

Tabela 2.2: Primerjava Arduino sistemov po velikosti pomnilnikov (Flash, SRAM, EEPROM).

plošče za Arduino Mega.

2.1.2 Merilna celica za težo - Vishay 1022 C3

Uporabili smo merilno celico s kapaciteto 100 kg (kar zadostuje za naše potrebe, saj povprečen panj AŽ tehta med 40 in 80 kg). Odstopanje meritev je 0.02%, kar je več kot zadovoljivo [8].



Slika 2.1: Merilna celica za težo - Vishay 1022 C3.

2.1.3 Senzor za temperaturo in vlago Hunwei DHT21

V testnem obdobju smo uporabili senzor temperature in vlage DHT11, vendar pa se je izkazalo, da tehtnica v primeru, da jo imamo zunaj tudi pozimi, odpove. Njegov razpon je namreč med 0 in 50 stopinjami Celzija in razponom vlažnosti med 20 in 90%.

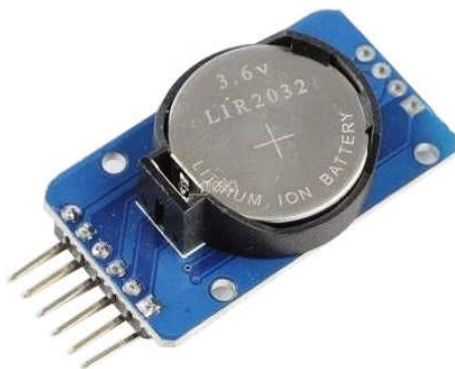
Zato smo se odločili uporabiti senzor DHT21 [9] z razponom temperature med -40 in 80 stopinjami Celzija in razponom relativne vlažnosti med 0 in 100%.



Slika 2.2: Senzor za vlago in temperaturo DHT21.

2.1.4 Modul za realni čas (RTC) - DS3231

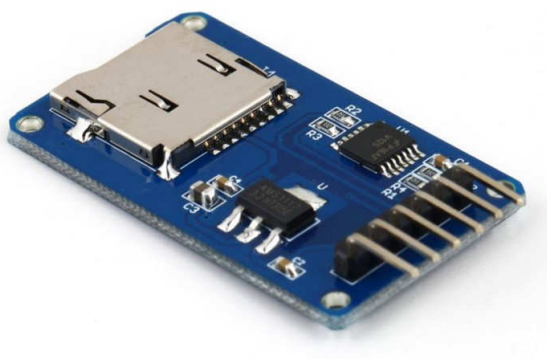
Arduino sam po sebi nima možnosti, ki bi nam omogočala pošiljanje SMS sporočila ob točno določeni uri. Zato smo se odločili, da uporabimo modul, ki nam to omogoča. Med mnogimi moduli RTC(Real clock timer) je bil izbran RTC modul DS3231 [10], saj nam omogoča več možnosti pri nadaljnjem delu (varčevanje energije).



Slika 2.3: RTC modul DS3231.

2.1.5 Modul za pomnilniške MicroSD kartice

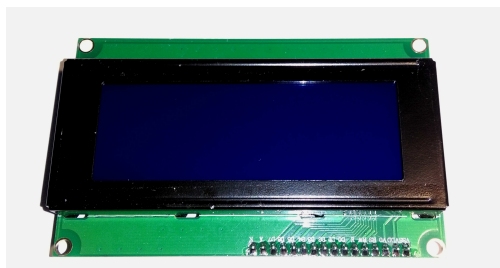
V želji po dodatnem shranjevanju podatkov kar lokalno v napravi smo se odločili, da v našo napravo vgradimo še MicroSD modul [12]. Uporabili ga bomo za shranjevanje podatkov na pomnilniško kartico in bomo tako imeli shranjeno dodatno kopijo naših podatkov.



Slika 2.4: MicroSD modul.

2.1.6 LCD prikazovalnik - Blue Serial I2C 20X4

Včasih želimo pogledati stanje na tehtnici tudi v čebelnjaku. Zato potrebujemo za prikaz podatkov LCD prikazovalnik [11]. Za naš projekt smo izbrali štiri vrstični lcd prikazovalnik. Vsaka vrstica ima na voljo 20 znakov za zapis željenih podatkov, kar popolnoma zadostuje za nas.



Slika 2.5: LCD prikazovalnik - Blue Serial I2C 20X4.

2.1.7 GSM ščit - GSM nadgradnja osnovne plošče Arduino

GSM ščit ([13]) je nadgradnja osnovne plošče Arduino in omogoča napravi, da deluje kot mobilni telefon. Osnovni funkciji GSM ščita sta pošiljanje/prejemanje SMS sporočila ter klicanje in prejemanje klicev. Za prejemanje SMS sporočil potrebujemo dodatno le še ekran, kamor se sporočilo izpiše, za pošiljanje pa tudi tipkovnico. V primeru, da želimo opravljati klice, pa potrebujemo tudi zvočnik in mikrofona, tipkovnico (številčnico) ter tipki za sprejem in zavrnitev klica.

V našem primeru smo uporabili le del funkcij, ki nam jih gradnik ponuja. Uporabili smo funkcije za pošiljanje SMS sporočil in shranjevanje podatkov v podatkovno bazo preko podatkovnega prenosa.



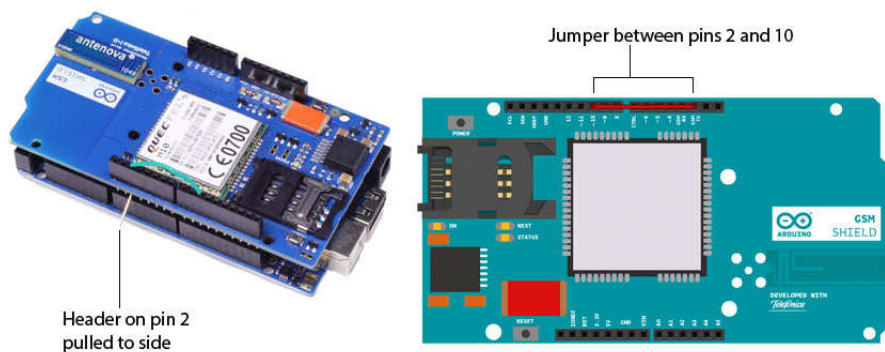
Slika 2.6: GSM ščit podjetja Telefonica za osnovne plošče Arduino [13].

2.2 Povezovanje gradnikov v sistem

Povezovanje Arduina z gradniki je dokaj enostavno. Arduino ima lepo označene priključne nožice in če imamo starter kit, lahko vezavo naredimo še lažje s priloženimi žicami, saj imajo konektorje prilagojene obema stranema (tako osnovni plošči kot dodatnim gradnikom).

2.2.1 Vezava modula GSM

GSM ščit oziroma modul je narejen tako, da ga samo postavimo na osnovno ploščo. Konektorji se namreč popolnoma prilegajo eden na drugega, vsaj če uporabljamo Arduino UNO. V našem primeru pa smo morali zakriviti pin 2 na GSM ščitu (na spodnji strani) in ga na zgornji povezati s pinom 10.

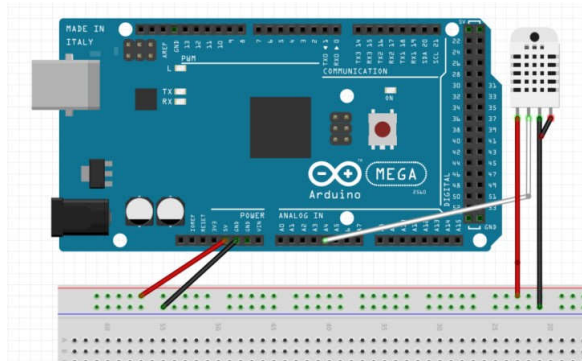


Slika 2.7: Vezava osnovne plošče Arduino in GSM ščita [14].

2.2.2 Vezava senzorja za vlago in temperaturo DHT21

Senzor za vlago DHT21 je enostaven za priključitev, saj zanj potrebujemo le tri povezave: priključka napetosti + in - ter podatkovno povezavo na Arduino. V našem primeru je to digitalni priključek D4.

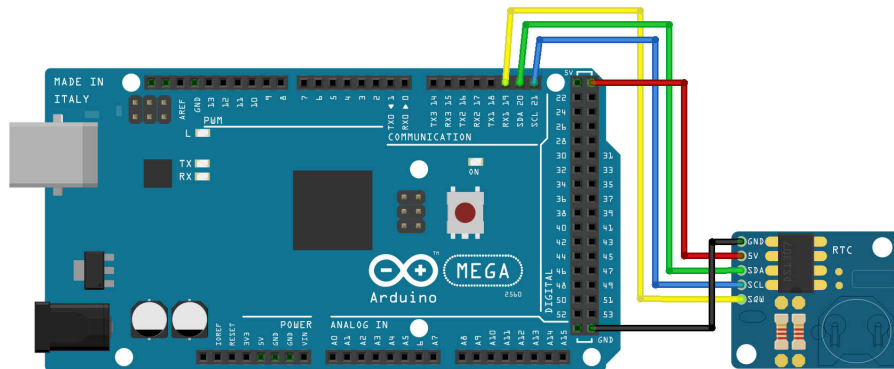
Ker merjenje temperature in vlage v notranjosti čebelnjaka enostavno nima smisla, smo dali senzor na daljše žice in ga namestili na sprednji del le tega pod strop. Senzor je tako večino dneva v senci in nam zato kaže bolj realne rezultate.



Slika 2.8: Vezava Arduino Mega in senzorja za temperaturo in vlago DHT21.

2.2.3 Vezava modula za realni čas - DS3231

Modul RTC povežemo z Arduinom tako, da povežemo ozemljitev in napajanje z napajanjem in ozemljitvijo na Arduino plošči. Nato povežemo priključka SDA in SCK na prava dva priključka na plošči.



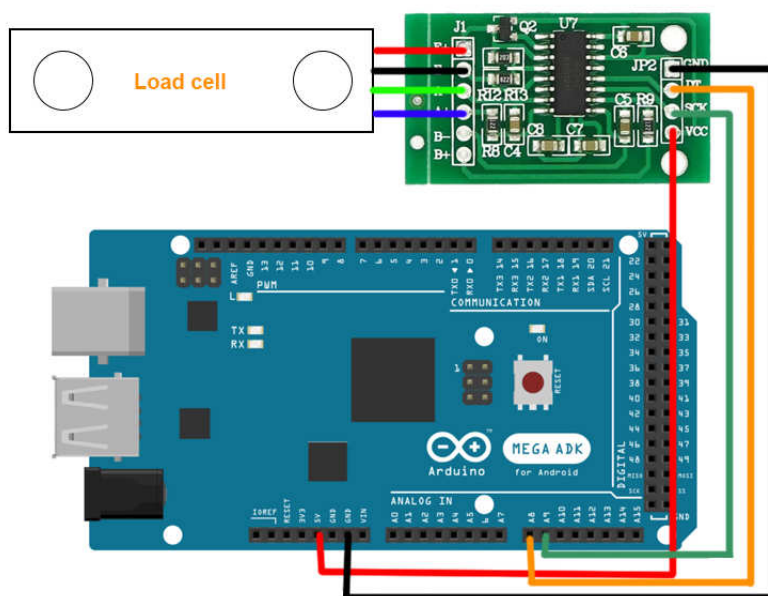
Slika 2.9: Vezava osnovne plošče Arduino in modula za realni čas - DS3231.

2.2.4 Vezava merilne celice teže Vishay 1022 C3

Merilne celice ne moremo direktno priključiti na Arduino modul. Razlog je v prenizkih napetostih, ki jih merilna celica oddaja. Za zajem nižjih napetosti smo uporabili poseben analogni digitalni pretvornik - modul HX711 [17].

Modul nam omogoča enostavno priključitev do dveh merilnih celic na eno napravo in s tem merjenje na dveh mestih (v našem primeru lahko v enem čebelnjaku tehtamo dva panja in s tem dobimo natančnejše podatke).

Priključimo ga tako, da na eno stran priključimo napajanje in ozemljitev iz Arduino plošče ter dva analogna vhoda (A0 - oranžna in A1 - zelena), prek katerih beremo podatke iz merilne celice. Na drugi strani pa priključimo žici za napajanje (rdeča) in ozemljitev (črna) iz merilne celice ter pozitiven (svetlo zelena) in negativen (modra) izhod iz celice.

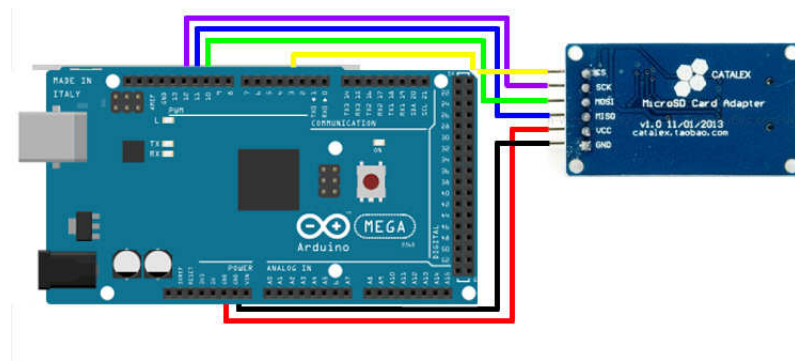


Slika 2.10: Vezava osnovne plošče Arduino, modula HX711 in merilne celice za težo.

Da pa lahko merilna celica sploh deluje jo moramo postaviti v ogrodje. To ogrodje je sestavljeno iz dveh kovinskih delov v obliki črke H. Ogrodje je velikosti 400 mm x 400 mm in je dovolj veliko za montažo pod AŽ panj. Ogrodje je skupaj z merilno celico visoko 50 mm.

2.2.5 Vezava modula za shranjevanje na microSD

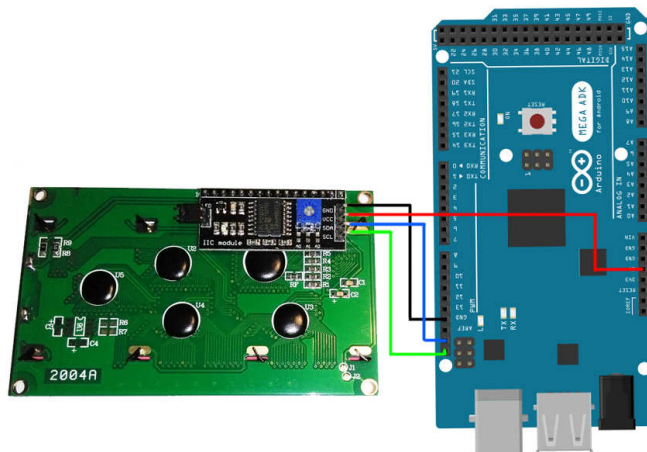
GSM ščit, ki smo ga uporabili za naš projekt, na Arduino Mega uporablja priključek 10 za "CS" (Chip Select), tako kot tudi microSD modul. MicroSD modul je bilo zato potrebno priključiti na priključke osnovne plošče in ne na priključke GSM ščita. Arduino Mega ima CS priključek na številki 53. Za priključitev microSD modula potrebujemo +5V, GND (ozemljitev) MOSI (priključek 50), MISO (priključek 51) SCK (priključek 52), priključek 53 mora biti prazen in ga nastavimo v programski kodi, kot CS (Chip Select).



Slika 2.11: Vezava osnovne plošče Arduino in microSD modula.

2.2.6 Vezava prikazovalnika LCD Blue Serial I2C 20X4

Vezava LCD prikazovalnika je zaradi I2C gradnika enostavna in minimalna. Le ta omogoča, da LCD prikazovalnik priključimo s samo štirimi priključki. V nasprotnem primeru, bi jih potrebovali veliko več. Tako potrebujemo le priključka za napajanje in ozemljitev ter priključka za prenos podatkov (SDA in SCL). LCD gradnik omogoča tudi enostavno regulacijo kontrasta prikazovalnika, potrebujemo le manjši izvijač.



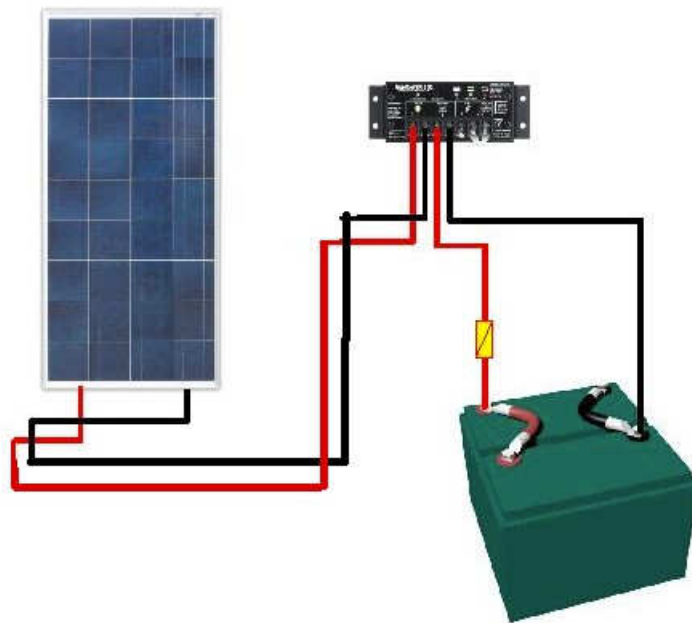
Slika 2.12: Vezava osnovne plošče Arduino in LCD prikazovalnika.

2.2.7 Napajanje

Arduino je v kombinaciji z GSM ščitom in senzorji zelo potraten glede porabe električne energije. Meritve so pokazale, da ob polni moči porabi 0,2A toka. Zato za napajanje skrbi akumulator s kapaciteto 15Ah in 20W sončnimi celicami.

V kolikor vklopimo na Arduino možnost spanja in s tem posledično varčevanje s energijo, lahko tak akumulator zdrži tudi več kot teden dni brez sonca, ki skrbi za polnjenje le tega.

Sončnih celic ne moremo povezati neposredno z akumulatorjem. Uporabimo vmesni člen, solarni regulator Epsolar LS0512 5A 12V [15]. Regulator ima vgrajeno zaščitno funkcijo pred preveliko napolnjenostjo ali izpraznjenostjo akumulatorja. Povežemo žici za napetost na regulator in iz regulatorja na akumulator. Porabnik je priključen na regulator, saj tako onemogočimo popolno izpraznitev akumulatorja.



Slika 2.13: Primer vezave sončnih celic preko ustreznega krmilnika na akumulator [16].

Poglavje 3

Programska oprema

Programska koda je sestavljena iz šestih sklopov:

- inicializacija spremenljivk in uvoz knjižnic
- funkcije za merjenje teže
- funkcije za temperaturo in vlago
- funkcije za pošiljanje SMS sporočil
- funkcije za shranjevanje podatkov na SD kartico
- funkcije za shranjevanje podatkov v MySQL podatkovno bazo.

V prvem sklopu kode uvozimo knjižnice za delo z

- GSM modulom
- modulom za prikaz podatkov na LCD prikazovalniku
- modulom za realni čas DS3231
- merilno celico Vishay 1022 C3
- modulom za temperaturo in vlago DHT21
- modulom za microSD kartico

Obenem pa tu še inicializiramo vse potrebne spremenljivke in funkcije, ki so potrebne za nadaljnje delo. Vso to kodo napišemo v inicializacijskem delu programa - funkcija "setup()"

Vseh nadaljnjih 5 sklopov programske kode je zapisanih v neskončni zanki v funkciji "loop()".

3.1 Branje podatkov iz merilne celice ter senzorja za vlago in temperaturo

Vsak dan ob 21. uri preberemo vrednost s senzorja za težo, jo zapišemo v spremenljivko in odštejemo trenutno težo od teže, ki smo jo shranili prejšnji dan ob 21. uri.

Sledi branje senzorja za vlago in temperaturo, rezultate shranimo v spremenljivke, da jih pozneje lahko enostavneje uporabimo.

3.2 Funkciji pošiljanja SMS sporočil

Funkcija pošiljanja SMS ima dve možnosti - SMS na zahtevo in SMS ob določeni uri. Koda je v obeh primerih zelo podobna, razlika je v pogojnem stavku in začetku funkcije.

3.2.1 SMS na zahtevo

Pri možnosti SMS na zahtevo, preberemo SMS in shranimo številko pošiljatelja. Nato jo uporabimo za pošiljanje SMS sporočila. V pogojnem stavku preverjamo, ali je SMS sporočilo prispelo. Ko je pogoj izpolnjen se izvede pošiljanje SMS sporočila.

```
//preverimo ce je smo prejeli SMS
if (sms.available()) {
    //dolocimo na katero stevilko se bo poslal SMS
    sms.remoteNumber(senderNumber, 10);
    //zacetek SMS sporocila
    sms.beginSMS(senderNumber);
```



```
// z ukazom sms.print(""); vnasamo vsebino
SMS sporočila
sms.print("Zacetek_sms");
//...
//zakljucimo sms z ukazom sms.endSMS();
sms.endSMS();
};
```

3.2.2 SMS poslan ob določeni uri

Pri možnosti SMS ob določeni uri pa je številka, kamor se SMS pošlje, že vnaprej določena. Pogojni stavek čaka, da je ura 21:00 in takrat izvede pošiljanje SMS sporočila.

```
//ce je ura 21:00 potem posljemo podatke.
if (now.hour() == 21 && now.minute() == 00){
    char Number[] = "031123456";
    sms.beginSMS(Number);
        \\vsebina sms-a
    }
}
```

3.3 Shranjevanje na microSD kartico

Funkcija, ki omogoča shranjevanje na microSD kartico. Beleženje začnemo s pogojnim stavkom, ki preverja ali datoteka obstaja, nato sledi zapisovanje v datoteko s funkcijo print, ko končamo zapisovanje, datoteko zapremo.

```
if (myFile) {
    myFile.print("vsebina_datoteke");
    _myFile.println();
    _File.close();
}
```

3.4 Shranjevanje v podatkovno bazo

Za shranjevanje v podatkovno bazo MySQL na oddaljenem strežniku potrebujemo povezavo z internetom, kar preverimo s pogojnim stavkom

```
if (client.connect(server , port)){
    Serial.println("connected");
    // Make a HTTP request:
    client.print("GET_");
    client.print("/cebelca/submit.php?datum=");
    client.print(now.day());
    client.print(".");
    //sledijo ostali podatki
    client.println("_HTTP/1.1");
    client.print("Host:_");
    client.println(server);
    client.println("Connection:_close");
    client.println();
}
```

Ob uspešni povezavi s strežnikom nato prenesemo podatke na spletno stran s končnico .php, kjer vse podatke podamo v URL naslovu in se ti nato na strežniku obdelajo in shranijo v primerni obliki v podatkovno bazo.

3.5 Varčevanje z energijo

Zadnji del kode skrbi za spanje Arduina, da nam ne porabi preveč energije. Za spanje je poskrbljeno z uporabo knjižnice LowPower in s klicem funkcije funkcije LowPower.powerDown :

```
\\posljemo arduino v sleep mode. ce zelimo daljsi
\\cas (v tem primeru spi 8 sekund) potem to vrstico
\\ kopiramo večkrat zapored.
LowPower.powerDown(SLEEP\_8S , ADC\_OFF , BOD\_OFF);"
```

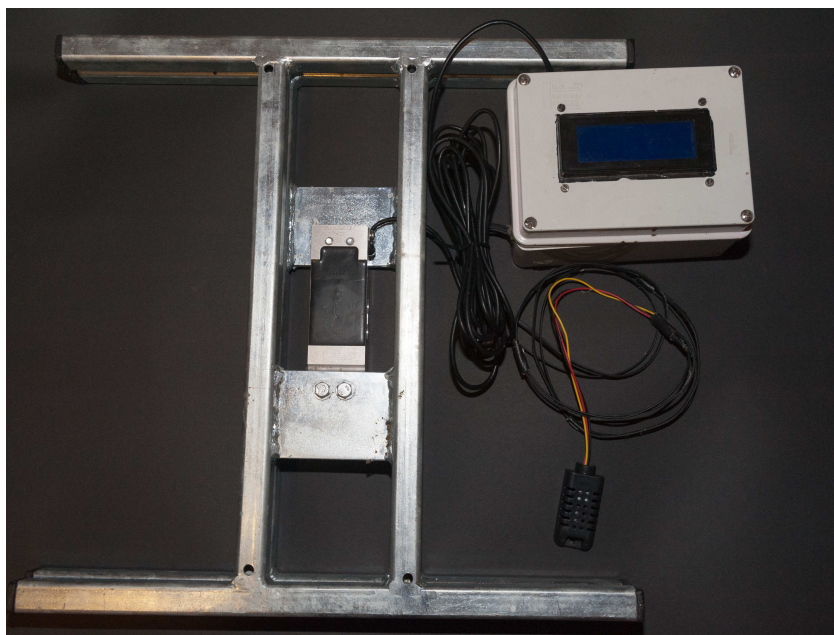
Funkcija postavi napravo v spanje za 8 sekund, za daljše spanje kličemo funkcijo večkrat zapored.

Poglavje 4

Uporaba končnega izdelka v praksi

Naš izdelek je sestavljen iz dveh delov - merilne celice z ogrodjem in vgrajenim sistemom ter zaslonom LCD, ki sta povezana v celoto s pomočjo žic. Merilno celico v ogrodju namestimo tako, da panj dvignemo in pod njega vstavimo ogrodje. Panj pustimo dvignjen, priključimo tehtnico na napajanje in na sosednji panj postavimo vgrajen sistem.

Počakamo, da se program naloži in nam izpiše vse potrebne podatke. Ko program na tehtnici naložen (na zaslonu se nam izpišejo podatki o teži, vlagi, temperaturi in datum), spustimo panj na ogrodje z merilno celico in pri tem pazimo, da se panj ne dotika nobenega drugega predmeta. To preverimo tako, da ga zazibamo in le ta se mora pri tem prosto zazibati (v primeru, da se dotika predmeta, se ne bo zazibal, ampak bo vanj trčil). Če se zaziba, potem je vse pripravljeno, v nasprotnem primeru odpravimo težavo.



Slika 4.1: Končni izdelek - Pametna elektronska tehtnica s senzorjem vlage in temperature.

Na tehtnici se nam izpiše trenutna teža panja in tehtnica je nastavljena za uporabo. Paziti moramo le na to, da bo na SIM kartici vedno dovolj dobroimetja za pošiljanje SMS sporočil.

4.1 Montaža v čebelnjak

Naš izdelek je dokaj enostaven za montažo. Panj najprej dvignemo, ko je panj dvignjen pod njega vstavimo merilno celico v ogrodju. Preden panj spustimo na tehcnico priključimo napajanje na vgrajen sistem. Počakamo, da se program zažene (ko na ekranu vidimo podatke o teži in vlagi je tehcnica pripravljena na uporabo). Počasi spustimo panj na ogrodje in poskrbimo, da se panj ne dotika nobenega predmeta v okolici. V primeru, da bi se dotikal, podatki niso pravilni. To najlažje preverimo tako, da ga na rahlo zazibamo. Če se ne dotakne ničesar in se lepo zaziba je vse pripravljeno.



Slika 4.2: Pravilno vgrajena tehcnica pod AŽ panjem.

4.2 Spletni prikaz shranjenih podatkov

V želji, da bi imel uporabnik čim boljši pregled nad dogajanjem v čebelnjaku čez celo leto smo izdelali spletno stran, na kateri bo imel uporabnik zabeležene vse meritve, saj je pregledovanje prek SMS sporočil zamudno in nepregledno.

Uporabnik bo imel ob naročilu tehtnice izbiro, kdaj oziroma kolikokrat dnevno se bodo podatki beležili na spletno stran. Privzeta nastavitve na tehtnici je pošiljanje podatkov v bazo ob 21. uri, saj so takrat podatki najbolj uporabni.

Podatke pridobivamo okrog 21. ure zato, ker so takrat čebele že v panju in tako dobimo najbolj optimalne podatke. Meritve moramo opravljati vedno ob istem času, saj le na ta način dobimo uporabne podatke. Le te lahko opravljamo kadarkoli (primer popoldan ob 15. uri), vendar taki podatki niso dovolj dobri. V primeru da meritve opravljamo sredi dneva se lahko zgodi, da je lepo vreme in so vse čebele zunaj ali pa dežuje in so vse v panju. Če pa meritve izvajamo ob 21. uri pa so čebele po biološki uri vedno v panju.

V osnovni verziji spletne strani je uporabniku prikazanih zadnjih 15 meritev v tabeli, graf pa prikazuje meritve čez celotno leto. Na začetku koledarskega leta bo evidenca prejšnjega leta shranjena v arhiv, graf pa se bo začel ustvarjati znova. Tako bo imel uporabnik pregled po letih in tako možnost boljše analize svojega dela pri čebelah.

Za izris grafa smo uporabili knjižnico "Google charts", kjer beremo podatke iz podatkovne baze.

```
//izpisemo podatke iz podatkovne baze.  
$rows = array();  
$flag = true;  
$table = array();  
$table['cols'] = array(  
    array('label' => 'Datum', 'type' => 'string'),  
    array('label' => 'Teza', 'type' => 'number'),  
    array('label' => 'Temperatura', 'type' => 'number'),  
    array('label' => 'Vlaga', 'type' => 'number'),  
    array('label' => 'Razlika', 'type' => 'number')
```

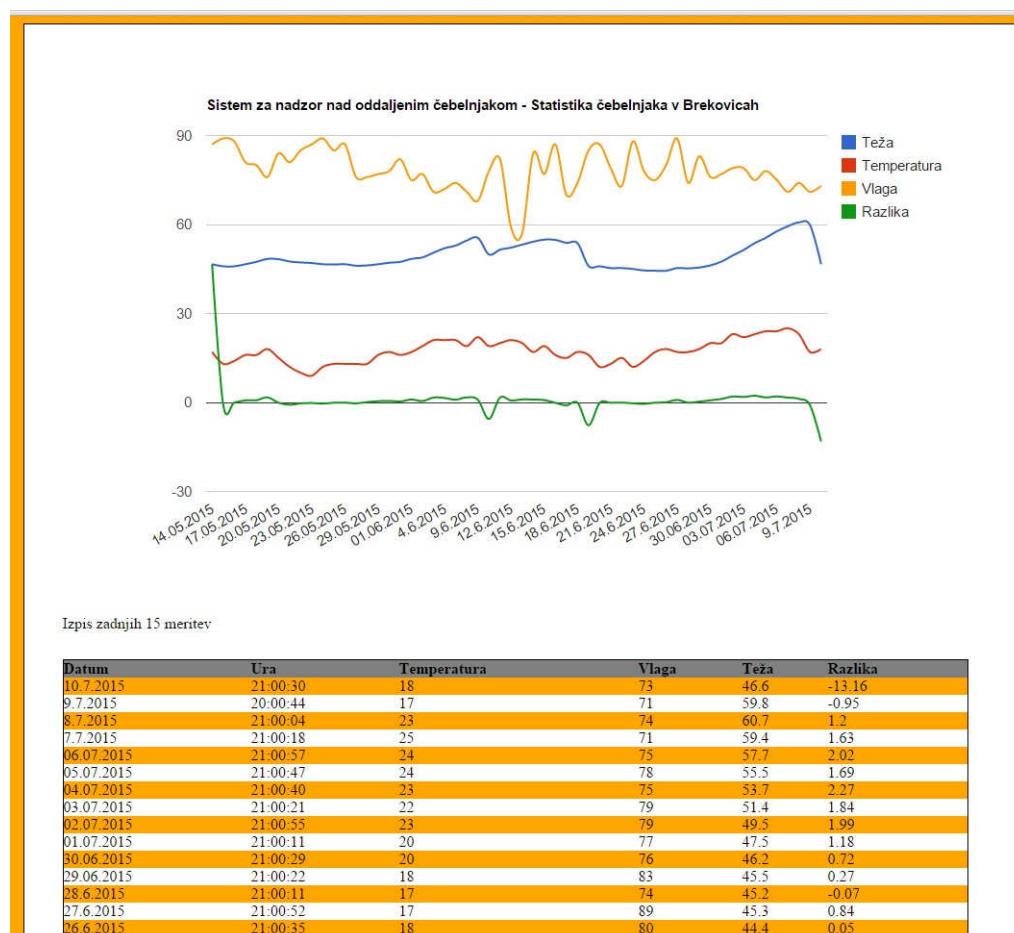
```
);  
$rows = array();  
while($r = mysql_fetch_assoc($sth)) {  
    $temp = array();  
    $temp[] = array('v' => (string) $r['datum']);  
    $temp[] = array('v' => (float) $r['teza']);  
    $temp[] = array('v' => (int) $r['temperatura']);  
    $temp[] = array('v' => (int) $r['vlaga']);  
    $temp[] = array('v' => (float) $r['razlika']);  
    $rows[] = array('c' => $temp);  
}  
  
$table['rows'] = $rows;  
$jsonTable = json_encode($table);  
?>  
//skripta ki generira izpis grafa  
<script type="text/javascript">  
    google.load("visualization", "1", {packages:  
        ["corechart"]});  
    google.setOnLoadCallback(drawChart);  
    function drawChart() {  
        var data = new google.visualization.DataTable  
        (<?=$jsonTable?>);  
        var options = {  
            title: 'Sistem za nadzor cebelnjaka',  
            curveType: 'function',  
            explorer: { actions: ['dragToZoom',  
                'rightClickToReset'],  
                axis: 'both' },  
            width: 1024,  
            height: 600  
        };  
    };
```

```
var chart = new google.visualization.LineChart
(document.getElementById( 'chart_div '));
chart.draw(data , options );
}
</script>

//to kodo vpisemo tja kamor zelimo postaviti graf.
<div id="wrapper">
<div id="chart_div"></div></div>
```

Vse to omogoča tehnika, ki pošilja podatke prek HTTP zahteve na strežnik, kjer se podatki shranijo v podatkovno bazo in nam tako omogočajo nadaljnjo statistično obdelavo.

Kot dodatno so izpisani dnevi, ko je bila razlika v teži največja v pozitivno (torej medenje ali hranjenje čebel) in v negativno (točenje ali večji poseg v panj).



Slika 4.3: Prikaz spletne strani uporabniku.

4.3 Testiranje sistema v praksi

V začetni fazi projekta se je pojavila težava z velikostjo pomnilnika za shranjevanje programske kode na Arduino Uno (le 32 kb), kar pa ne zadostuje za vse funkcije, ki smo jih implementirali. Rešitev našega problema je Arduino Mega, ki ima 256 kb pomnilnika.

V času testiranja (od 6. 5. 2015 do 11. 9. 2015) je tehtnica delovala večinoma tako, kot je bilo predvideno. Vendar pa ni šlo brez težav, po nekaj dneh delovanja je tehtnica namreč programsko "zamrznila" in jo je bilo potrebno resetirati. Napaka, ki je povzročila neskončno zanko je bila zahteva za stanje na mobilnem računu, saj je vrnila napako namesto stanja, zato je bil odstranjen ta del kode (saj za nas ni

ključnega pomena) in tehtnica je bila ponovno zagnana.

Po približno treh tednih se tehtnica ponovno ni odzivala, tokrat smo po krajšem iskanju napako našli v microSD modulu, ki se je med delovanjem kar sam od sebe ugašal. Začasna rešitev je bila izklop microSD modula. Modul je bil nato doma testiran, napaka odpravljena in nato je bil vrnjen nazaj po nekaj dneh v uporabo.

Po slabih treh mesecih je zaradi nepazljivosti uporabnika zmanjkalo sredstev na mobiračunu, ker v kodi do takrat še ni bilo implementirano, kaj se zgodi ob takem primeru, je bil potreben ponovni zagon naprave. Če uporabnik v času nedelovanja pošlje SMS na napravo, ta po zagonu ne deluje pravilno in tudi ob ponovnem zagonu se težava ne odpravi, zato je potrebno tehtnico priklopiti na računalnik in prenesti vse poslane SMS in nato ponovno zagnati sistem.

Ena od težav je tudi ta, da sistem ob vsakem zagonu nastavi novo "taro". Torej je potrebno ob vsakem ponovnem zagonu panj dvigniti s tehtnice, počakati, da se sistem naloži in ga nato ponovno spustiti na merilno celico. Problem smo rešili tako, da smo v spominsko kartico zabeležili vrednost, ki določa težo tehtnice brez obremenitve. Tako lahko tehtnico ponovno zaženemo in imamo pravo težo, brez da bi bilo potrebno razbremeniti merilno celico.

Poglavje 5

Sklepne ugotovitve

V tem diplomskem delu je bil izdelan vgrajen sistem z elektronsko tehniko, ki čebelarju omogoča lažji nadzor nad oddaljenim čebelnjakom. Izdelek olajša delo uporabniku (v našem primeru čebelarju), saj mu prihrani pot do lokacije in s tem posledično tudi čas, ki ga lahko koristneje uporabi.

Vsakodnevne meritve in njihovo beleženje uporabniku pomagajo do lažje evidence in hitrejšega odkrivanja začetka oziroma konca medenja. Tako lahko uporabnik, ko pride na lokacijo, takoj ve, kaj je potrebno storiti (ali je to točenje medu ob koncu paše ali hranjenje v brez pašnem obdobju, itd).

Izdelek se vsekakor da še veliko izboljšati. Nadgraditi bi bilo potrebno predvsem omogočanje uporabniku prek SMS sporočil spreminjati nastavitve (kdaj in na katero številko se pošiljajo SMS sporočila, itd). Nekateri uporabniki ne želijo prejemati SMS sporočil vsak dan, ampak le na zahtevo. V tem primeru je potrebno še implementirati funkcijo, ki bo poslala uporabniku razliko teže od njegovega zadnjega poslanega SMS ter aktualno temperaturo in vlago.

Tehnico smo testirali doma v domačem čebelarstvu in če odštejemo nekaj težav s prenizkim stanjem na SIM kartici je tehnika delovala dobro. Precej nam je olajšala delo, saj smo prek SMS sporočil lahko vsak dan spremljali stanje v čebelnjaku in tako vedeli, kaj se v njem dogaja.

Cena trenutne tehnice je okrog 300€ brez DDV. Vendar glede na to, kako je mogoče tehniko še optimizirati in nadgraditi, se lahko cena spreminja od 250 pa vse tja do 500€. Funkcije tehnice so tako rekoč neomejene. Cene konkurenčnih tehnic se gibajo med 400€ in 2000€ z DDV.

S tako tehtnico čebelar prihrani kar nekaj časa. Vzemimo primer, da imamo čebelnjak 20 km stran. Torej potrebujemo 15 minut vožnje v vsako smer, dodatnih 10-20 minut da vozilo parkiramo, odklenemo čebelnjak in odčitamo podatke. Vse skupaj čebelarju vzame 40-50 minut časa. Predvidimo, da take obiske izvaja trikrat tedensko. To počne 30 tednov, kar pomeni 90 obiskov. Če vse te obiske preračunamo v ure, dobimo približno 75 ur ali dobre 3 dni. Vsak obisk pomeni tudi 40 kilometrov, vemo pa da gorivo ni brezplačno. Vsak obisk čebelnjaka stane čebelarja še približno dodatne 4€, kar na leto znese približno 360€. To je strošek za en čebelnjak. Tako ima čebelar pokrite stroške tehtnice v enem letu.

Literatura

- [1] Microsoft.(2016).Vse o mobilni širokopasovni povezavi [Online]. Dostopno na: <http://windows.microsoft.com/sl-si/windows-8/mobile-broadband-from-start-to-finish>
- [2] Eldema Srečko Lavrič. (2015). SMS Čebelar [online]. Dostopno na: <http://www.eldema.si/sl/>
- [3] Danilo Bedek.(2014). Telemetrija čebelnjaka [Online]. Dostopno na: <http://team-pikica.si/telemetrija-cebelnjaka/>.
- [4] Armes. (2015). Čebelarska tehnica TCM-13 [online]. Dostopno na: http://www.ames.si/cat/products/cebelarska_tehnica/l:2
- [5] Neznani avtor. (2015) Arnia remote hive monitoring [Online]. Dostopno na: <http://www.arnia.co.uk/>
- [6] Neznani avtor. (2013). Arduino vs Raspberry Pi Comparison [Online]. Dostopno na: <http://codeluino.com/tutorials/arduino-vs-raspberry-pi/>
- [7] Neznani avtor. (2014). Arduino MEGA 2560 [Online]. Dostopno na: <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>
- [8] Vishay Precision Group. (2014). Single-Point Aluminum Load Cell [Online]. Dostopno na <http://www.vishaypg.com/docs/12007/1022.pdf>
- [9] Hunwei. (2014). HM2301 Digital-output humidity and temperature sensor [Online]. Dostopno na: <http://www.electrodragon.com/w/images/6/6f/DHT21.pdf>

-
- [10] Maxim Integrated Products, Inc. (2015). DS3231 Extremely Accurate I2C-Integrated RTC/TCXO/Crystal [Online]. Dostopno na: <http://datasheets.maximintegrated.com/en/ds/DS3231.pdf>
- [11] Saint Smart. (2015). SainSmart Blue Serial IIC/I2C/TWI 2004 20x4 LCD Module Shield for Arduino Raspberry Pi [Online]. Dostopno na: <http://www.sainsmart.com/sainsmart-iic-i2c-twi-serial-2004-20x4-lcd-module-shield-for-arduino-uno-mega-r3.html>
- [12] Catalex. (2013). MicroSD Card Adapter [Online]. Dostopno na <https://drive.google.com/file/d/0B4oKoy5QnlfMZU15TkxHVWpNN28/view>
- [13] Neznani avtor. (2015). Arduino GSM Shield by Telefonica [Online]. Dostopno na: <http://ubidots.com/docs/devices/GSMTelefonica.html#arduino-gsm-shield-by-telefonica>
- [14] Neznani avtor. (2015). How to use the Arduino GSM shield with Arduino Leonardo, Arduino Yun, and Arduino Mega [Online]. Dostopno na: <https://www.arduino.cc/en/Guide/GSMShieldLeonardoMega>
- [15] Epsolar. (2013). LS0512 — Solar Charge Controller [Online]. Dostopno na: <http://midsummerenergy.co.uk/pdfs/ep5a.pdf>
- [16] Keoghs Marine. (2015). Wiring diagram [Online]. Dostopno na: <https://www.keoghsmarine.com.au/image/data/Keoghs%20Marine%20Parallel%20Solar%20Panel%20Wiring%20Diagram.jpg>
- [17] Bodge. (2014). An Arduino library to interface the Avia Semiconductor HX711 24-Bit Analog-to-Digital Converter (ADC) for Weight Scales [Online]. Dostopno na: <https://github.com/bodge/HX711>